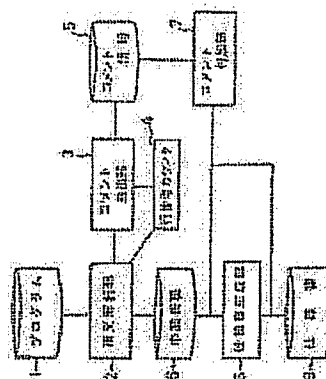


**AUTOMATIC GENERATION SYSTEM FOR PROGRAM SPECIFICATIONS****Publication number:** JP4055938 (A)**Publication date:** 1992-02-24**Inventor(s):** NAOTA SHIGEKI**Applicant(s):** FUJITSU LTD**Classification:****- international:** G06F9/06; G06F9/06; (IPC1-7): G06F9/06**- European:****Application number:** JP19900166442 19900625**Priority number(s):** JP19900166442 19900625**Abstract of JP 4055938 (A)**

**PURPOSE:** To automatically generate the easy-to-understand specifications from a program by extracting previously the comment information described in the program in response to a line No. and adding the comment information to the specifications to be automatically generated.

**CONSTITUTION:** The comment information 5 is extracted from the program inputted through a comment extracting part 3 in response to a line No. at syntactic analysis. Then a comment addition part 7 applies a syntactic analysis to the input program and adds the information 5 to an intermediate expression 6 generated with addition of the line No. based on the line No. of the extracted information 5. A specifications generating part 8 generates automatically the specifications with comments based on the expression 6 including the information 5. Thus the easy-to-understand specifications can be automatically obtained.



Data supplied from the esp@cenet database — Worldwide

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A) 平4-55938

⑤ Int. Cl.<sup>5</sup>

G 06 F 9/06

識別記号

4 3 0 E

庁内整理番号

7927-5B

⑬ 公開 平成4年(1992)2月24日

審査請求 未請求 請求項の数 2 (全10頁)

⑭ 発明の名称 プログラム仕様書自動生成方式

⑯ 特 願 平2-166442

⑰ 出 願 平2(1990)6月25日

⑱ 発 明 者 直 田 繁 樹 神奈川県川崎市中原区上小田中1015番地 富士通株式会社  
内

⑲ 出 願 人 富 士 通 株 式 会 社 神奈川県川崎市中原区上小田中1015番地

⑳ 代 理 人 弁 理 士 岡 田 守 弘

明 細 書

1. 発明の名称

プログラム仕様書自動生成方式

2. 特許請求の範囲

(1) プログラムからコメント付の仕様書を自動生成するプログラム仕様書自動生成方式において、  
入力されたプログラムから構文解析時にコメント情報(5)を行番号に対応づけて抽出するコメント抽出部(3)と、

入力されたプログラムを構文解析して行番号を付加して生成した中間表現(6)に対して、上記抽出したコメント情報(5)の行番号をもとに当該コメント情報(5)を付加するコメント付加部(7)とを備え、

このコメント情報(5)を付加した中間表現をもとにコメント付の仕様書(9)を生成するように構成したことを特徴とするプログラム仕様書自動生成方式。

(2) プログラムからコメント付の仕様書を自動

生成するプログラム仕様書自動生成方式において、

入力されたプログラムから構文解析時にコメント情報(5)を行番号に対応づけて抽出するコメント抽出部(3)と、

入力されたプログラムを構文解析して行番号を付加して生成した中間表現(6)をもとに生成した行番号付の仕様情報に対して、上記抽出したコメント情報(5)の行番号をもとに当該コメント情報(5)を付加し、仕様書(9)を生成するコメント付加部(7)とを備え、

このコメント付加部(7)によって生成したコメント付の仕様書(9)を出力するように構成したことを特徴とするプログラム仕様書自動生成方式。

3. 発明の詳細な説明

(概要)

プログラムからコメント付の仕様書を自動生成するプログラム仕様書自動生成方式に関し、

プログラム中に記述されたコメント情報を抽出しておき、自動生成する仕様書に反映し、より分

かり易い仕様書を自動生成することを目的とし、

入力されたプログラムから構文解析時にコメント情報を行番号に対応づけて抽出するコメント抽出部と、入力されたプログラムを構文解析して行番号を付加して生成した中間表現に対して、上記抽出したコメント情報の行番号をもとに当該コメント情報を付加するコメント付加部とを備え、このコメント情報を付加した中間表現をもとにコメント付の仕様書を生成するように構成する。

#### 〔産業上の利用分野〕

本発明は、プログラムからコメント付の仕様書を自動生成するプログラム仕様書自動生成方式に関するものである。銀行システムのような膨大な既存のプログラムの保守作業を容易にするために、仕様書を電子的に管理し、参照検索を容易にすることが要求されている。既存のプログラムの仕様書の多くは手書きであり、電子的に管理できる仕様書を作るために、プログラムから仕様情報を抽出して仕様書を自動生成することが望まれている。

の仕様書を生成する対象のプログラムである。

コメント抽出部3は、構文解析時にプログラム1からコメント情報5を抽出するものである。

中間表現6は、プログラム1を構文解析して生成した行番号付の中間表現である。

コメント付加部7は、中間表現6にコメント情報5を付加したり、中間表現6から生成した仕様情報にコメント情報5を付加したりするものである。

仕様書生成部8は、コメント付の中間表現6から仕様書を生成したり、中間表現6から行番号付の仕様情報を生成したりするものである。

#### 〔作用〕

本発明は、第1図に示すように、コメント抽出部3が入力されたプログラムから構文解析時にコメント情報5を行番号に対応づけて抽出し、コメント付加部7が入力されたプログラムを構文解析して行番号を付加して生成した中間表現6に対して、抽出したコメント情報5の行番号をもとに当

#### 〔従来の技術〕

従来の仕様書生成方式は、プログラムの構文を解析する際に直接関係のないコメント情報を削除していた。しかし、コメントは、プログラムでは記述しない説明的な情報が書かれており、プログラムを理解し、保守するために重要なものである。

#### 〔発明が解決しようとする課題〕

従来の方式は、コメントを削除していたため、自動生成される仕様書が人間に分かり難いものになってしまうという問題があった。

本発明は、プログラム中に記述されたコメント情報を抽出しておき、自動生成する仕様書に反映し、より分かり易い仕様書を自動生成することを目的としている。

#### 〔課題を解決するための手段〕

第1図は、本発明の原理構成図を示す。

第1図において、プログラム1は、コメント付

該コメント情報5を付加し、このコメント情報5を付加した中間表現6をもとに仕様書生成部8がコメント付の仕様書を自動生成するようにしている。

また、コメント抽出部3が入力されたプログラムから構文解析時にコメント情報5を行番号に対応づけて抽出し、仕様書生成部8が入力されたプログラムを構文解析して行番号を付加して生成した中間表現6をもとに行番号付の仕様情報を生成し、コメント付加部7がコメント情報5の行番号をもとに当該コメント情報5を仕様情報に付加し、仕様書を自動生成するようにしている。

従って、プログラム中に記述されたコメント情報を行番号に対応づけて抽出しておき、自動生成する仕様書に反映することにより、より分かり易い仕様書を自動生成することが可能となる。

#### 〔実施例〕

まず、第1図から第9図を用いて本発明の1実施例の構成および動作を順次詳細に説明する。

第1図において、構文解析部2は、プログラム1を構文解析し、行番号を付加した中間表現6を生成するものである。

行番号カウンタ4は、構文解析中のプログラム1の行番号をカウントするものである。

第2図は、本発明の1実施例構成図を示す。

第2図において、プログラム1は、ソースプログラムであって、例えばCOBOL言語で記述した第5図プログラム例である。

コメント抽出部3は、構文解析部2から呼び出され、後述する第3図フローチャートに従い、例えば第5図プログラム例から第7図コメント情報5を抽出するものである。

行番号カウンタ4は、コメント抽出部3がコメント情報5をプログラム1から抽出するときに当該プログラム1の行番号をカウントするものである。

中間表現(1)6-1は、プログラム1を構文解析して生成した行番号を付加した中間表現(1)である。例えば第5図プログラム例から第6図行番号を付

加した中間表現例である。

コメント付加部7は、中間表現(1)6-1に対して、コメント情報5の行番号をもとに当該コメント情報5を付加し、中間表現(2)6-2を生成するものである。例えば第6図中間表現例(中間表現(1))に、第7図コメント情報例を付加し、第8図中間表現例(中間表現(2))を生成するものである。

中間表現(2)6-2は、中間表現(1)6-1にコメント情報5を付加した中間表現である。

仕様書生成部8は、コメント情報5を付加した中間表現(2)6-2から仕様書9を生成するものである。例えば第8図中間表現例(中間表現(2))から第9図仕様書例を生成するものである。

仕様書9は、プログラム1から生成したコメント情報付の仕様書であって、例えば第9図仕様書例である。

次に、第3図フローチャートに示す順序に従い、第2図コメント抽出部3の処理を、第5図ないし第7図を用いて詳細に説明する。

第3図において、①は、

行番号カウンタ←1

コメントフラグ   off

記号フラグ       off

コメント←空文字列

記号←空文字列

に初期設定する。

②は、プログラムを1文字読み込む。これは、例えば第5図プログラム例の若い行番号からプログラムを1文字読み込む。

③は、改行コードか否かを判別する。YESの場合には、④で行番号カウンタ4を1増やす(プログラム1から読み込んだ1文字が改行であったので、次の行番号に設定するために、行番号カウンタの内容を1増す)。一方、NOの場合には、⑤を行う。

⑥は、コメントフラグがonか否かを判別する。YESの場合(現在処理中の1文字がコメントの場合)には、⑦ないし⑨を行う。NOの場合(現在処理中の1文字がコメントでない場合)には、⑩でこの1文字がコメント開始文字(例えば第5

図コメント開始文字"\*)"であるか否かを判別し、YESのときに⑪で

コメントに1文字追加

コメントフラグ   on

コメント行番号←行番号カウンタ(現在の行番号)

とし、⑪以降を繰り返し行い、NOのときに⑪以降を繰り返し行う。一方、NOの場合には、コメント抽出を行うことなく、⑪ないし⑨で記号抽出を行う。

⑫は、コメント終了文字(例えばコメント終了文字"\*)"か否かを判別する。YESの場合には、⑬でコメント行番号とコメントを組にしてコメント情報として保存する。更に、

コメントフラグ   off

コメント←空文字列

に設定し、⑪以降を繰り返し行う。一方、NOの場合には、⑬でコメントに1文字、追加し、⑪以降を繰り返し行う。

以上の⑪ないし⑬の処理によって、プログラム

1 (例えば第5図プログラム例) からコメント情報5 (例えば第7図コメント情報例中の文1と記載した部分のコメント情報) を抽出する。

第3図において、④は、記号構成文字か否かを判別する。YESの場合には、④で記号フラグがonか否かを判別し、YESのときに④で記号に1文字追加して④以降を繰り返し行い、NOのときに④で記号フラグをon、記号に1文字追加して④以降を繰り返し行い。一方、NOの場合には、④で記号フラグがonか否かを判別し、YESのときに④で記号を返し (例えば第5図プログラム例中の0002行の"EVALUATE"をまとめて返し) て終了し、NOのときに④以降を繰り返し行い。

以上の④ないし⑤の処理によって、プログラム1 (例えば第5図プログラム例) から第6図中間表現例を生成するために必要な記号 (例えば0002行からEVALUATE、KOZANOCKなど) を抽出する。

次に、第4図フローチャートに示す順序に従い、構文解析処理を、第5図および第6図を用いて詳細に説明する。

加し、第8図中間表現例 (中間表現②) を生成するときの処理を説明する。

(1) コメント情報を若い方から1つ取り出す。これは、第7図コメント情報例から若い方の例えば行番号0001の"\*\*\*口座番号の入力チェック\*\*\*"を取り出す。

(2) 中間表現①をサーチし、近い行番号を見つける。これは、①で取り出した"\*\*\*口座番号の入力チェック\*\*\*"の行番号0001に近い行番号ここでは第6図中間表現①の④の位置から行番号"0002"を見つける。

(3) 見つけた行番号を消し、コメントを中間表現に付加する。これは、②で見つけた第6図④の行番号0002を消し、ここにコメント"\*\*\*口座番号の入力チェック\*\*\*"を付加し、第8図④に示すように中間表現②にする。

以下同様に、第7図行番号0005の"\*\*\*入力なしエラー\*\*\*"、行番号0010の"\*\*\*その他のエラー\*\*\*"について、第6図④、⑤をそれぞれ見つけ、第8図④、⑤に示すようにコメントをそれ

第4図において、⑥は、記号読み込みを行う。これは、第3図⑤でプログラム中から抽出した記号 (例えば第5図プログラム例の0002行の"EVALUATE"、"KOZANOCK"など) を読み込む。

⑦は、ファイルの終わりか否かを判別する。YESの場合には、終了する。NOの場合には、⑧を行う。

⑧は、文法にしたがって文を認識する。

⑨は、1文受理したか否かを判別する。YESの場合には、⑨で行番号を付与した中間表現を生成する。これは、例えば1文である第5図プログラム例の全体を受理した場合、第6図中間表現例を生成する。一方、NOの場合には、⑥以降を繰り返し行い。

以上の④ないし⑨の構文解析処理を、第3図④でプログラム (例えば第5図プログラム例) から抽出した記号について行い、中間表現① 6-1 (例えば第6図中間表現例) を生成する。

次に、第7図コメント情報例をもとに、第6図中間表現例 (中間表現①) にコメント情報を付

それ付加した中間表現②にする

第9図は、本発明に係るコメント付で出力した仕様書例を示す。これは、第8図中間表現例 (中間表現②) から仕様書を自動生成したものである。ここで、第9図④、⑥、⑦のコメントは、第8図中間表現例 (中間表現②) の④、⑥、⑦に対応している。

次に、第10図ないし第15図を用いて本発明の他の実施例構成の動作を詳細に説明する。

第10図は、本発明の他の実施例構成図を示す。ここで、プログラム1、構文解析部2、コメント抽出部3、行番号カウンタ4、コメント情報5、中間表現① 6-1は、第2図構成と同じであるので、説明を省略する。

第10図において、仕様書生成部8-1は、中間表現① 6-1から行番号付の仕様情報9-1を生成するものである。例えば第6図中間表現① 6-1から行番号付の第12図仕様情報例を生成するものである。

仕様情報9-1は、中間表現① 6-1から生成

した行番号付の仕様情報（例えば第12図仕様情報例）である。

コメント付加部7-1は、仕様情報9-1に、コメント情報5中の行番号をもとに当該コメントを付加し、仕様書（例えば第9図仕様書例）を生成するものである。

第11図フローチャートに示す順序に従い、仕様書生成部8-1の処理を説明する。

第11図において、④は、中間表現を1文読み込む。これは、既述したように、例えば第5図プログラム例から生成した第6図中間表現例（中間表現①）から1文を読み込む。ここでは、第6図中間表現例が1文であるので、当該第6図中間表現例を全部読み込む。

④は、中間表現ファイルが空か否かを判別する。YESの場合には、中間表現の全ての文の処理を終了したので、④で記録した補足生成部（例えば第9図中の補足記述）を出力し、終了する。NOの場合には、④以降を行う。

④は、辞書（第13図）を用いて英字名を日本

語名に置き換える。例えば第6図英字名"KOZANOC K"を日本語名"口座番号判定コード"に置き換える。

④は、文に変換ルール（第14図）を適用する分岐文の場合は個々の処理に含まれる文のも再帰的に変換ルールを適用する。

④は、変換した文が登録パターン（第15図）のパターン部とマッチする部分を含むかを調べる。YESの場合には、④以降を行う。NOの場合には、④で変換結果を出力する（第12図仕様情報例を出力する）。

④は、マッチした部分を登録パターンの結果部で置き換える。これは、④で変換した文が第15図登録パターンのパターン部とマッチする部分を含むと判別されたので、マッチした部分を登録パターンの結果部で置き換える。

④は、登録パターンが既に使われているかを調べる。YESの場合には、何もしなく、④を行う。NOの場合には、④で補足生成部（第12図補足記述）を記憶し、④を行う。

以上の処理について、第6図中間表現例について、第13図辞書によって日本語に置き換えた後、第14図変換ルール例および第15図登録パターン例を適用すると、第12図仕様情報例が得られる。具体的に説明すると、

①：第6図①に、第14図①を適用し、第12図①を生成する。この例は、第11図④NOとなり、第15図登録パターン例と比較しない。

②：第6図②に、第14図②（2つ）を適用し、第12図②を生成する。この例も、第11図④NO（第15図パターン部のマッチする部分を含まないで、④NO）となり、第15図登録パターン例と比較しない。

③：第6図③に、第14図③（4つ）を適用し、第12図③を生成する。この例は、第11図④YESとなり、第15図登録パターン例のパターン部で、エラーメッセージ番号=1（=?1）、エラー項目番号=口座番号（=?）、およびセクションERR-SECTION 実行、にそれぞれマッチし、結果部で、"処理1（1、口座番号）"にそれぞれ

置き換えて生成する。また、補足生成部も生成する。

第12図は、本発明に係る行番号付の仕様情報例を示す。これは、既述したように、第6図行番号付の中間表現①から第11図フローチャートに示す処理によって生成した行番号付の仕様情報である。①ないし④は、第6図①ないし④から生成したものである。

第13図は、本発明に係る英字名日本語名対応辞書例を示す。これは、第6図中間表現①中の英字名を、日本語に置き換える辞書である。

第14図は、本発明に係る変換ルール例を示す。これは、第6図中間表現①を第12図仕様情報例に変換するための変換ルールである。図中の①ないし④は、第6図中間表現①の①ないし④に適用する変換ルールを示す。

第15図は、本発明に係る登録パターン例を示す。これは、第14図変換ルール例を中間表現①に適用し、第12図仕様情報例を生成するためのものである。

以上の処理によって生成した第12図仕様情報例について、第10図コメント付加部7-1が第7図コメント情報例の行番号をもとに、近い行番号の位置に当該コメント情報例を埋め込んで付加し、第9図仕様書例を生成する。

#### 〔発明の効果〕

以上説明したように、本発明によれば、プログラム中に記述されたコメント情報を行番号に対応づけて抽出しておき、自動生成する仕様書に付加する構成を採用しているため、より分かり易い仕様書をプログラムから自動生成することができる。

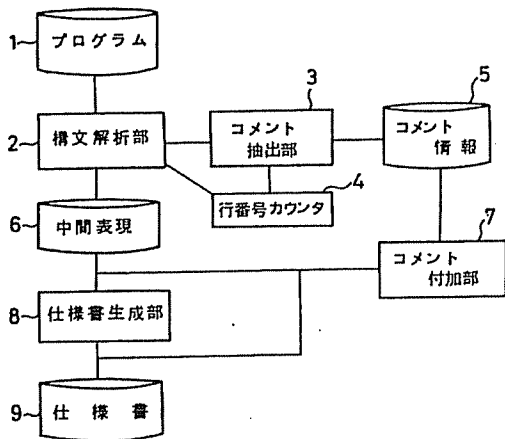
#### 4. 図面の簡単な説明

第1図は本発明の原理構成図、第2図は本発明の1実施例構成図、第3図は本発明に係るコメント抽出フローチャート、第4図は本発明に係る構文解析フローチャート、第5図はプログラム例、第6図は本発明に係る構文解析によって生成した中間表現例、第7図は本発明に係る保存したコメ

ント情報例、第8図は本発明に係るコメントを付加した中間表現例、第9図は本発明に係るコメント付で出力した仕様書例、第10図は本発明の他の実施例構成図、第11図は本発明に係る仕様書生成部の処理フローチャート、第12図は本発明に係る行番号付の仕様情報例、第13図は本発明に係る英字名日本語名対応辞書例、第14図は本発明に係る変換ルール例、第15図は本発明に係る登録パターン例を示す。

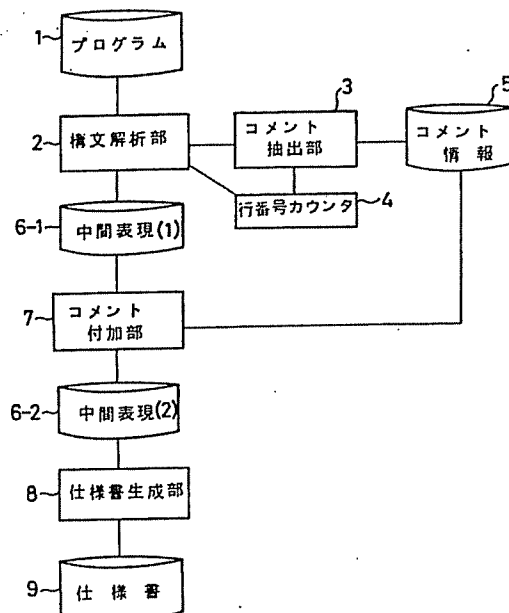
図中、1はプログラム、2は構文解析部、3はコメント抽出部、4は行番号カウンタ、5はコメント情報、6は中間表現、6-1は中間表現(1)、6-2は中間表現(2)、7、7-1はコメント付加部、8、8-1は仕様書生成部、9は仕様書、9-1は仕様情報を表す。

特許出願人 富士通株式会社  
代理人弁理士 岡田 守弘



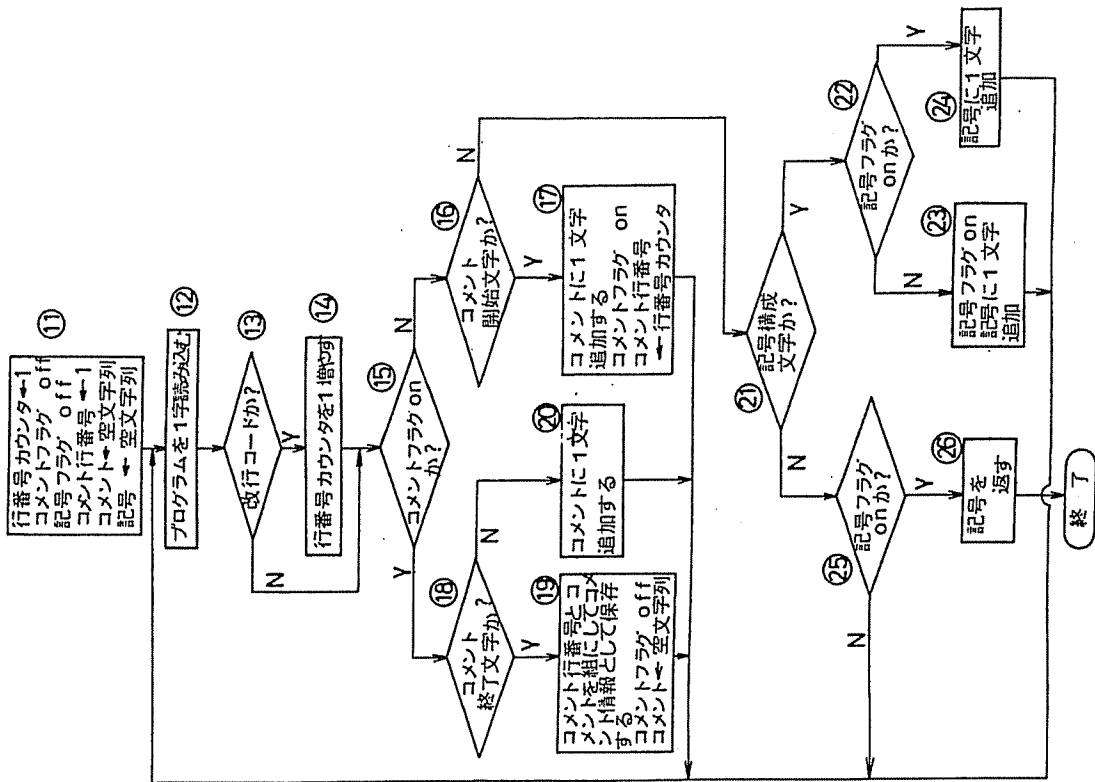
本発明の原理構成図

第 1 図



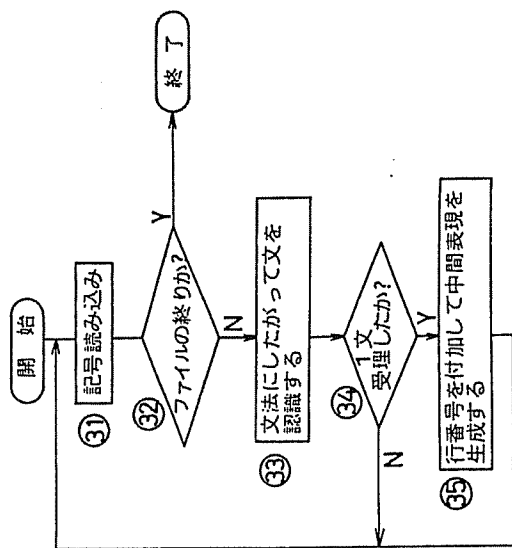
本発明の1実施例構成図

第 2 図



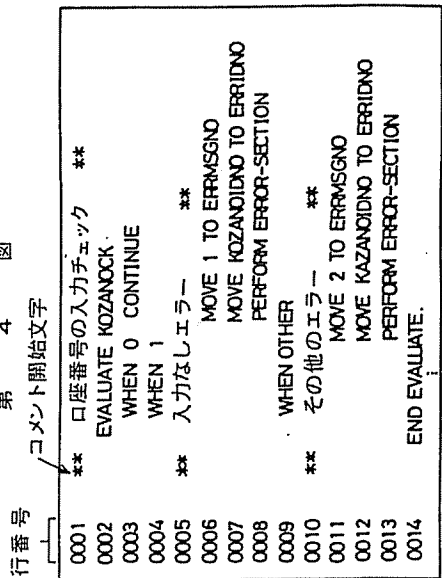
本発明に係るコメント抽出フローチャート

第 3 図



本発明に係る構文解析フローチャート

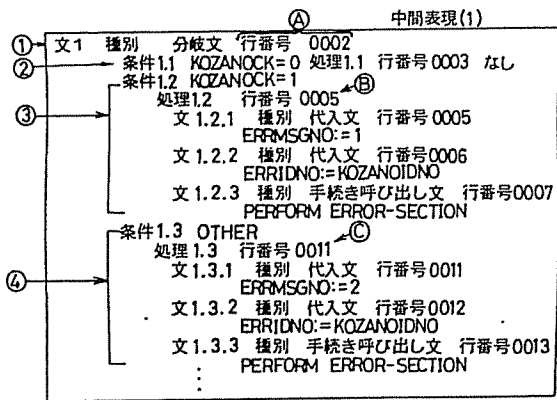
第 4 図



プログラム例

第 5 図





本発明に係る構文解析によって生成した中間表現例

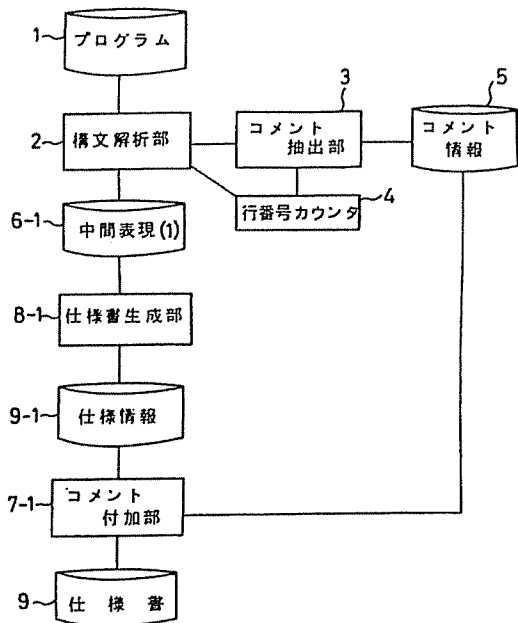
第 6 図

文1

0001	xx	口座番号の入力チェック	xx
0005	xx	入力なしエラー	xx
0010	xx	その他のエラー	xx
0015	xx	利率コードの入力チェック	xx
0019	xx	コード値エラー	xx
0025	xx	入力なしエラー	xx
0030	xx	その他のエラー	xx
...			

本発明に係る保存したコメント情報例

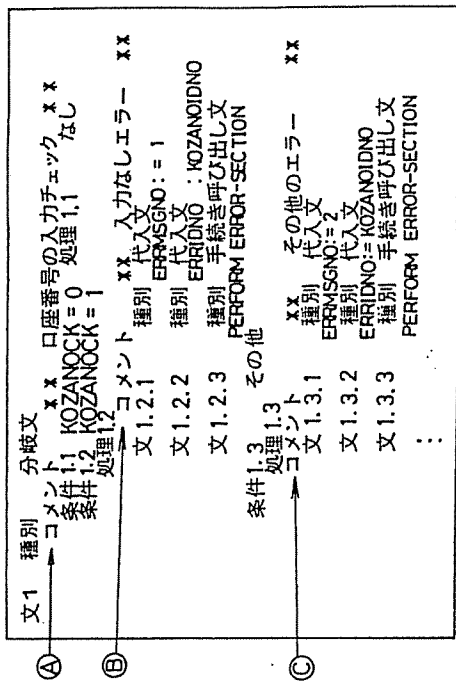
第 7 図



本発明の他の実施例構成図

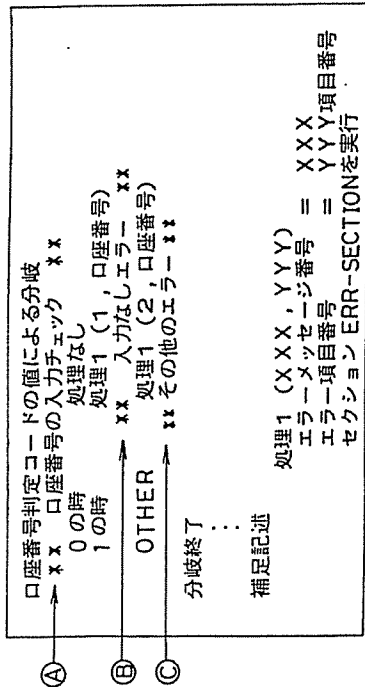
第 10 図

中間表現(2)



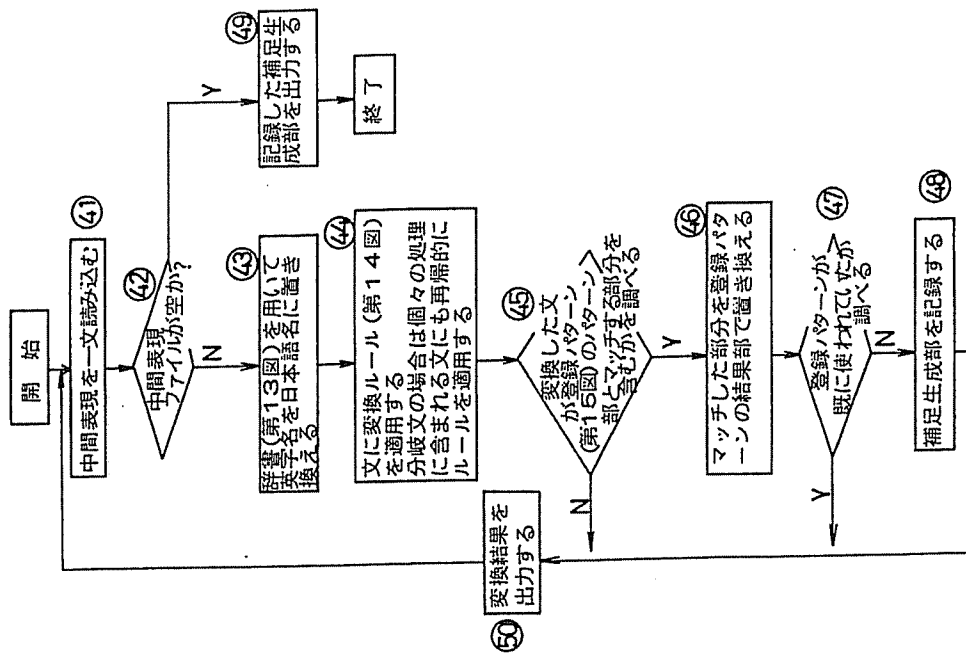
本発明に係るコメントを付加した中間表現例

第 8 図



本発明に係るコメント付で出力した仕様書例

第 9 図



本発明に係る仕様書生成部の処理フローチャート

第 11 図

①	口座番号判定コードの値による分岐	行番号 0002
②	0の時	処理なし
③	1の時	処理1 (1, 口座番号)
④	その他	処理1 (2, 口座番号)
	分岐終了	行番号 0011
	利率コード判定コードの値による分岐	行番号 0016
	0の時	利率コード = 0 ならば
		処理1 (3, 利率コード)
	1の時	処理1 (1, 利率コード)
	その他	処理1 (2, 利率コード)
	分岐終了	行番号 0031
⑤	補足記述	
⑥	処理1 (XXX, YYY)	
	エラーメッセージ番号 = XXX	
	エラー項目番号 = YYY項目番号	
	セクションERR-SECTIONを実行	

本発明に係る行番号付の仕様情報例

第 12 図

KOZANOCK	口座番号判定コード
KOZANOIDNO	口座番号項目番号
ZENZANCK	前残判定コード
ZENZANIDNO	前残項目番号
RIRITUCK	利率コード判定コード
RIRITU	利率コード
RIRITUIDNO	利率コード項目番号
ERRMSGNO	エラーメッセージ番号
ERRIDNO	エラー項目番号

本発明に係る英字名日本語名対応辞書例

第 13 図

ルール1  
文の種別が分岐文の場合  
1. 1 各条件部が、同一の変数の値の比較か、「OTHER」である時  
〔生成文〕  
変数「の値による分岐」 分岐文の行番号 ← ①  
条件1の右辺「の時」 個々の処理の変換結果 ← ②  
条件2の右辺「の時」 個々の処理の変換結果 ← ③  
「その他」 個々の処理の変換結果  
1. 2 それ以外  
〔生成文〕  
条件1 「ならば」 分岐文の行番号  
条件2 「ならば」 個々の処理中の文の変換結果  
： 個々の処理中の文の変換結果  
ルール2  
文の種別が代入文の場合  
〔生成文〕  
左辺「=」右辺 行番号 ← ②、③、③  
ルール3  
文の種別が手続き呼び出し文の場合  
〔生成文〕  
「セクション」セクション名「を実行」 行番号 ← ③  
この変換ルールで「」に囲まれた部分はその文字列を生成する。

本発明に係る変換ルール例

第 1 4 図

パターン部  
エラーメッセージ番号 = ? 1 行番号 ? 2  
エラー項目番号 = ? 3 項目番号 行番号 ? 4  
セクションERR-SECTIONを実行 行番号 ? 5  
結果部  
処理1 (? 1, ? 3) 行番号 ? 2  
補足生成部  
処理1 (XXX, YYY) XXX  
メッセージ番号 = XXX  
エラー項目番号 = YYY 項目番号  
セクションERR-SECTION を実行

本発明に係る登録パターン例

第 1 5 図